Cooperative Transmission for Wireless Networks using Mutual-Information Accumulation

Stark C. Draper, *Member, IEEE,* Lingjia Liu, *Member, IEEE,* Andreas F. Molisch, *Fellow, IEEE,* Jonathan S. Yedidia, *Senior Member, IEEE*

***Abstract*—Cooperation between the nodes of wireless multihop networks can increase communication reliability, reduce energy consumption, and decrease latency. The possible improvements are even greater when nodes perform mutual information accumulation. In this paper, we investigate resource allocation for unicast and multicast transmission in such networks. Given a network, a source, and a destination, our objective is to minimize end-to-end transmission delay under energy and bandwidth constraints. We provide an algorithm that determines which nodes should participate in forwarding the message and what resources (time, energy, bandwidth) should be allocated to each.**

Our approach factors into two sub-problems, each of which can be solved efficiently. For any *transmission order* we show that solving for the optimum resource allocation can be formulated as a linear programming problem. We then show that the transmission order can be improved systematically by swapping nodes based on the solution of the linear program. Solving a sequence of linear programs leads to a locally optimal solution in a very efficient manner. In comparison to the proposed cooperative routing solution, it is observed that conventional shortest path multihop routing typically incurs additional delays and energy expenditures on the order of 70%. Drawing inspiration from this first, centralized, algorithm, we also present two distributed algorithms. These algorithms require only local channel state information. Simulations indicate that they yield solutions about two to five percent less efficient than the centralized algorithm.

I. Introduction

Cooperative relaying is one of the most active research areas in wireless communications. The use of relays leads to improvements in energy efficiency, due to reduced node-to-node distances, and in robustness to fading, due to the increased number of possible transmission paths.

The most basic forms of relaying, as used, for example, in the Zigbee standard [1], route information along a single path, forwarding data packets from one node to the next in a manner akin to a bucket brigade. More sophisticated methods that require tight synchronization between nodes at the physical and media access control (MAC) layer lead to much larger performance gains; see, e.g., [2]–[6] and references therein.

At a high level, routing in wireless networks can be broken down into three distinct sets of issues. The first is the design of physical and MAC layer techniques for relaying information from one set of nodes to the next. The second is resource allocation, i.e., identifying what system resources (time, energy, bandwidth) should be allocated to each node. The third set of issues concerns complications such as interference between transmitters, mobility of nodes in the network, and changing channel conditions. We consider a simplified model that avoids the third set of issues. We assume channel conditions are fixed over the duration of communication, and that interference between transmissions can be neglected.

Our focus is on the remaining issues of optimal transmission of messages through the network. Most prior works that consider these coupled problems are based on physical layer techniques that either use virtual beamforming or energy accumulation. In virtual beamforming, transmitters adjust the amplitude and phases of signals to interfere constructively at the receiver [7]–[9]. In energy accumulation, multiple transmissions are combined non-coherently by receiving nodes through, for example, space-time or repetition coding [10]–[12].

In this paper we consider networks that employ *mutual-information accumulation* at the physical layer [13], [14], and the solution of the associated resource allocation problems. We concentrate on the unicast problem where all nodes work together to get a single message from a single source to a single destination node, though our framework also encompasses the multicast problem—a single source and multiple destinations. The solutions developed herein could find application in scenarios where delay and energy consumption are paramount and users are willing to work together. Examples include military or emergency services, wireless sensor networks, or possibly voluntary social networks.

The difference between energy accumulation and mutual information accumulation is most easily understood by considering binary signalling over a pair of independent erasure channels. Two cooperating transmitters wish to transmit a common message to a single destination. If the erasure probabilities are both p_e, and both transmitters use the same code, then each symbol will be erased with probability ${p_e}^2$. Therefore, $1 - {p_e}^2$ novel parity symbols are received, on average, per transmission. If, instead, the two transmitters use different codes, on average $2(1 - p_e)$ novel parity symbols (which exceeds $1 - {p_e}^2$) are received per transmission. The latter is mutual-information accumulation, while the former is an example of energy accumulation.

For Gaussian channels (or fading channels with decoder channel state information (CSI)) at low signal-to-noise ratios (SNRs), energy accumulation is nearly equivalent to mutual-information accumulation as capacity is approximately linear in SNR. However, as SNR increases, mutual-information accu-

This work was presented in part at the Int. Zurich Seminar Commun., Zurich, Switzerland, March 2008 [27] and in part at the IEEE Int. Conf. Commun., Beijing, China, May 2008 [28].

S. C. Draper is with the Dept. of Electrical and Computer Engineering, University of Wisconsin, Madison, WI 53706 (E-mail: sdraper@ece.wisc.edu).

L. Liu is with Dallas Telecomm. R&D Center, Samsung Electronics (E-mail: lliu@sta.samsung.com).

A. F. Molisch is with the Department of Electrical Engineering, University of Southern California, Los Angeles, CA, 90089, USA. (email: Andreas.Molisch@ieee.org).

J. S. Yedidia is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139 (Email: yedidia@merl.com).

This work was done while all authors were at Mitsubishi Electric Research Laboratories, Cambridge MA 02139.

mulation gives better results than either virtual beamforming or energy accumulation. Mutual information accumulation can be realized through the use of rateless codes, of which Fountain and Raptor codes [15]–[17] are examples.

The main contributions of the paper are threefold.

- First, we present a formulation of the relaying problem with mutual-information accumulation where the objective is to minimize end-to-end delay under various bandwidth and energy constraints.
- Second, under the assumption of centrally available CSI, we detail an iterative optimization method that is based on solving a sequence of linear programs (LPs). Each LP optimizes the resource allocation for a given "transmission order," which corresponds to the route taken by the message through the network. The resulting allocation is then used to update the order. The method proceeds iteratively and can find good routes very efficiently.
- Finally, taking inspiration from our centralized solution, we provide two distributed algorithms that require only local CSI. Simulations show that the resulting solutions require less than 5% additional energy for the same end-to-end delay as the centralized solution.

We have found little prior work investigating routing and resource allocation in networks using mutual-information accumulation. In [13], Castura and Mao considered mutual information accumulation for a single-relay network. Mutual information accumulation is also investigated in a limited way in [14], but network "flooding" is assumed where all nodes transmit all the time; this is not an optimum way of using energy. Regarding LP-based resource allocation solutions for ad-hoc networks, in [10], [11], Maric and Yates posed the resource allocation and routing problem as an LP, but the physical layer technique assumed is energy accumulation. However, the use of mutual information accumulation complicates and strengthens the inter-node dependencies, making it much more difficult to attack the optimal transmission problem. Another heuristic algorithm for routing with energy accumulation was proposed by Chen et.al. in [12]. In [18], Zhao and Valenti derived a heuristic algorithm for relaying information using hybrid ARQ (automatic repeat request), with mutual information accumulation *over time*. In contrast to our paper, however, Zhao and Valenti assume that when relay nodes transmit simultaneously, they send out the same signal.

An outline of the paper is as follows. We present the system model in Sec. II. We present and discuss illustrative results in Sec. III. The centralized routing and resource allocation algorithm, and its constituent parts, are developed in Sec. IV. In Sec. V we describe the two distributed algorithms. We provide details of simulation results in Sec. VI and conclude in Sec. VII. Proofs are provided in the appendix.

II. System model

We consider a unicast network consisting of $N+1$ nodes: the source, the destination, and $N-1$ relay nodes. The objective is to convey a data packet composed of B bits from source to destination in the minimum time under sum-energy and bandwidth constraints. [1] The relays may participate actively in packet transmission or may remain silent for the duration of communication, depending on the energy budget and/or system bandwidth budget. Relay nodes operate under a half-duplex constraint: they can either transmit or receive but cannot do both simultaneously. To simplify analysis we assume that a node's only significant energy expenditure lies in transmission; reception, decoding, and re-encoding entail no significant overhead. We note that this assumption can be relaxed within the framework presented.

The ith node operates at a fixed transmit power spectral density (PSD) P_i (joules/sec/Hz), uniform across its transmission band. The propagation channel between each pair of nodes is modeled as frequency-flat and block-fading, where the coherence time of the channel is larger than any considered transmission time of the encoded bits. The channel power gain between the ith and the kth nodes is denoted $h_{i,k}$. Under these assumptions, the spectral efficiency of data transmission from node i to node k can be expressed as [20]

$$C_{i,k} = \log_2\left[1+\frac{h_{i,k}P_iW_i}{N_0W_i}\right] = \log_2\left[1+\frac{h_{i,k}P_i}{N_0}\right]\frac{\text{bits}}{\text{sec}\cdot\text{Hz}}, \quad (1)$$

where $N_0/2$ denotes the PSD of the (white) noise process.

If node i is allocated the time-bandwidth product A_i sec-Hz for transmission, the potential information flow from node i to node k is $A_iC_{i,k}$ bits. Our first assumption is that nodes use codes that are ideal in the sense that they fully capture this potential flow, working at the Shannon limit at any rate. Nodes are further designed to use *independently generated* codes. This choice connects to our second assumption that, without any rate loss, a receiver can combine information flows from two or more transmitters. If, for example, a pair of transmitting nodes i and j are allocated time-bandwidth products A_i and A_j, respectively, our two assumptions mean that node k can decode as long as the mutual information accumulated by node k exceeds the message size, i.e.,

$$A_iC_{i,k} + A_jC_{j,k} \geq B. \quad (2)$$

The use of independently-generated codes is crucial to the mutual-information accumulation condition reflected in (2). If the *same* code were used by each transmitter, the receiver would get multiple looks at each codeword symbol. This is "energy-accumulation." By getting looks at different codes (generated from the same B information bits) the receiver accumulates mutual information rather than energy.

The two assumptions of ideal codes and mutual-information accumulation from multiple streams can most naturally be realized (albeit approximately) through the use of "fountain" (or "rateless") codes [21]. The rate-adaptive nature of fountain codes is an advantage for networks that operate with inaccurate CSI estimates, or that are unpredictably time-varying. Though not the emphasis of this paper, using rateless codes would thus also significantly improve the robustness of the cooperative networks described here. The non-ideal nature of existing implementations of fountain codes can be handled within our model without undue trouble by incorporating an overhead factor of $(1+\epsilon)$ into the right-hand side of (2); see [14] for further discussion.

The network also operates under bandwidth and energy constraints. We study the case where these resources are constrained on a per-node basis, and also the case where the constraints are imposed on the sum allocation across nodes.

[1]Multiple messages can be transmitted in parallel over (quasi-) orthogonal channels. See the discussion in [14] and [19].

Such constraints involve the A_i and the A_iP_i products. Full details will be provided in Section IV.

III. Motivation

In this section we illustrate the improvements made possible by combining mutual information accumulation with route optimization for a simple one-dimensional network. This model is amenable to closed-form analysis. We present these results prior to their full derivation in Section IV-E, so that readers can develop a sense of the possible improvement before delving into the full details of the algorithms and analysis.

The one-dimensional network we consider consists of $N+1$ nodes equally-spaced along the line segment $[0, D]$. The source node 0 is located at the origin and the destination node N is located at D. The channel power gain between two nodes, $i < j$, is proportional to $(d_{i,j})^{-2} = (N/D)^2(i-j)^{-2}$. As is fully developed in Section IV-E, under a system-wide sum-bandwidth constraint W_{T}, we can analytically solve for the transmission duration τ_{c} achieved by our cooperative protocol.

Consider the case where $P_i = P$ for all i. In this case the cooperative strategy that minimizes the transmission duration τ_{c} is for the source (node 0) to transmit long enough that node 1 can decode the message and then to stop transmitting. At that point node 1 starts to transmit (since it has received the packet) and its connectivity $C_{1,k} > C_{0,k}$ for $k > 1$ (since $P_i = P$ for all i and $d_{1,k} < d_{0,k}$). Thus it is better to allocate the full system bandwidth to node 1 rather than reserving some so that node 0 can continue to transmit. Subsequent transmissions continue until the next node in the chain decodes. Each transmission is shorter than the previous ones due to the mutual information already accumulated by nodes further down the chain during earlier nodes' transmissions.

For comparison we also solve for τ_{nc}, the transmission duration achieved by the best non-cooperative scheme where mutual-information accumulation is not performed. In this protocol each node listens only to a *single* transmission. Unlike the cooperative system in which all nodes participate, in this system the optimal route depends on P. When P is sufficiently low, the optimal route is the same as the cooperative one. As P increases, however, some relay nodes are skipped. And, when P is sufficiently large, the optimal (i.e., τ_{nc} minimizing) strategy is direct source to destination transmission.

The cooperative gain, defined as $\tau_{\mathrm{nc}}/\tau_{\mathrm{c}}$, is plotted in Fig. 1 for unit-spaced nodes ($D = 100$, $N = 100$, $B = 20$ nats) as a function of transmission power PW_{T}. The curve is piece-wise linear. The non-differentiable break points correspond to the powers at which the optimal non-cooperative (shortest-path) route changes. For example, for (roughly) $0 \le PW_{\mathrm{T}} \le 8$, all 100 nodes participate, for $8 \le PW_{\mathrm{T}} \le 24$, half the nodes participate, for $24 \le PW_{\mathrm{T}} \le 47$, one-third participate, for $47 \le PW_{\mathrm{T}} \le 78$, one-quarter participate and so forth.

As N approaches infinity, and P approaches zero, so that the product PN^2 stays small, we show in Section IV-E that the cooperative gain converges to $\pi^2/6 \simeq 1.64$. As can be seen by inspecting Fig. 1, the cooperative gain is greater at higher transmission PSDs.

Note that in this example since $P_i = P$ for all i and the sum-bandwidth is fixed, the energy expended by the cooperative and non-cooperative schemes is $\tau_{\mathrm{c}}PW_{\mathrm{T}}$ and $\tau_{\mathrm{nc}}PW_{\mathrm{T}}$, respectively. In this case the ratio $\tau_{\mathrm{c}}/\tau_{\mathrm{nc}}$ is the same as the ratio between the energy expended in the cooperative and non-cooperative cases.

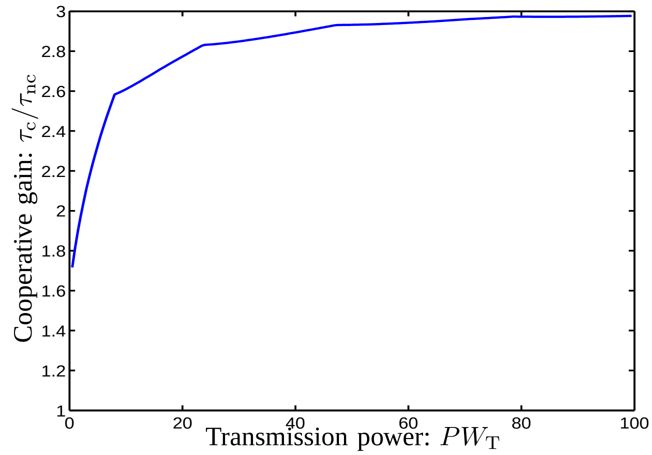

Fig. 1. Cooperative gain of the one dimensional network.

While the topology of this example is extremely simple, it illustrates two important facts. First, the use of mutual-information accumulation decreases latency and energy usage. Second, when mutual-information accumulation is used, the optimal route can be quite different from the optimum multi-hop route. These facts are also true for more complicated (and more practically relevant) two-dimensional networks.

IV. Centralized Algorithms

We now consider how to optimize system parameters to minimize delay for general networks. In Sec. IV-A we define the *transmission order* and, given a particular order, show that the resource allocation problem is an LP. In Sec. IV-B we use the solution of the LP to revise the transmission order to decrease end-to-end delay. Finally, in Sec. IV-C, we iterate between these two sub-problems.

A. Problem parametrization and LP-based resource allocation

Our parametrization of the problem revolves around the transmission order. We define the transmission order by starting with any ordering of the $N+1$ network nodes where the source node is the first node in the order. The transmission order is the sub-sequence that starts with the source node, always labelled 0, and ends with the destination node, always labelled L where $1 \le L \le N$. The transmission order indicates the order in which nodes are allowed to come on-line as transmitters. Since each node must decode before it can transmit, a node's position in the order puts constraints on the mutual information that that node must accumulate from earlier nodes in the order. As nodes $L+1, \ldots N$ never transmit (since they come on-line after the destination decodes), they are not considered part of the transmission order.

We denote the time at which node i decodes the message as T_i where $T_0 = 0$ and T_L is the duration of the source-to-destination transmission. Rather than the T_i we find it more useful to work with inter-node delays, Δ_i, where $\Delta_i = T_i - T_{i-1}$ for $1 \le i \le L$. Message transmission can be thought of as consisting of L phases. The ith phase is of duration Δ_i and at the *end* of the phase the first i nodes have all acquired the message.[2] We refer to each phase as a "time-slot". Time-slots are not of pre-set or equal lengths; rather their lengths are solved for in the LP stated next.

For a given transmission order we find the resource allocation minimizing end-to-end delay T_L,

$$T_L = \sum_{i=1}^{L} \Delta_i. \tag{3}$$

We minimize this linear objective function subject to the following constraints: (i) $\Delta_i \geq 0$ for all i, (ii) node i must decode by time $T_i = \sum_{l=1}^{i} \Delta_l$, (iii) the energy constraint(s), and (iv) the constraint(s) on the use of time and bandwidth. We state constraints (ii)–(iv) in turn.

First, there are L decoding constraints resulting from the nodes' positions in the transmission order

$$\sum_{i=0}^{k-1} \sum_{j=i+1}^{k} A_{i,j} C_{i,k} \geq B \quad \text{for all} \quad k \in \{1, 2, \ldots, L\}, \tag{4}$$

where

$$A_{i,j} \geq 0 \;\text{ for all }\; i \in \{0, 1, \ldots, L-1\},\; j \in \{1, 2, \ldots, L\}.$$

The $A_{i,j}$ are the time-bandwidth product (measured for example in $\text{seconds} \times \text{Hz}$) assigned to the ith node in the jth time slot. Recall that $C_{i,k}$ is the spectral efficiency (measured say in $\text{bits}/(\text{seconds} \times \text{Hz})$) of the channel connecting the ith transmitter to the kth receiver. Eq. (4) says that the total mutual information flow to the kth node in the transmission order must exceed B bits by the end of the kth time slot. Only the first $k-1$ nodes contribute to this sum.

Second, we consider constraints on energy and bandwidth. We consider here various possibilities for the types of constraints, including both *sum* constraints, applied to the sum-allocation across the network, and *per-node* constraints, applied to nodes individually.

1) Sum-energy constraint: A sum-energy constraint E_{T} is

$$\sum_{i=0}^{L-1} \sum_{j=1}^{L} A_{i,j} P_i = \sum_{i=0}^{L-1} \sum_{j=i+1}^{L} A_{i,j} P_i \leq E_{\mathrm{T}}. \tag{5}$$

where the equality holds because $A_{i,j} = 0$ for $j \leq i$ since node i has not decoded until the end of slot i and therefore can only transmit in slots $i+1, \ldots, L$.

2) Per-node energy constraint: An energy constraint E_i applied to node i is

$$\sum_{j=i+1}^{L} A_{i,j} P_i \leq E_i \quad \text{for all} \quad i \in \{1, 2, \ldots, L\}. \tag{6}$$

3) Sum-bandwidth constraint: A sum-bandwidth constraint W_{T} takes the form

$$\sum_{i=0}^{j-1} A_{i,j} \leq \Delta_j W_{\mathrm{T}} \quad \text{for all} \quad j \in \{1, 2, \ldots, L\}. \tag{7}$$

[2]In fact, as will become more clear when we discuss finding the best transmission order, additional nodes may have already decoded. But the first i node are guaranteed to have already decoded.

4) Per-node bandwidth constraint: If the system bandwidth is divided into parallel channels, which each can be allocated at most a single transmitter at any given time, we impose bandwidth constraints on a per-node basis. In this case, instead of the L constraints in (7) we get L^2 constraints:

$$A_{i,j} \leq \Delta_j W_i \quad \text{for all} \quad \begin{array}{l} i \in \{0, 1, \ldots L-1\} \\ j \in \{1, 2, \ldots, L\} \end{array}. \tag{8}$$

Regarding the sum-bandwidth constraint, several aspects of (7) are worth noting. First, the specific time-bandwidth allocation to each node *within* each transmission slot need not be specified. Since fading is modeled as block-fading and frequency-flat, each transmitter is agnostic as to what is its exact time-bandwidth allocation, i.e., degrees-of-freedom are treated like a fluid; only the allocated time-bandwidth product is important. We assume that each nodes is able to use optimally whatever region of the spectrum is allocated to it for transmission.

Because the degrees-of-freedom are treated as a fluid, the optimal solution under a sum-bandwidth constraint can always be implemented by scheduling just one node to transmit at any given instant. In time slot j we allocate the whole bandwidth to node i for duration of $A_{i,j}/W_{\mathrm{T}}$ sec. The ordering of transmissions within a time slot is immaterial since only at the end of the time slot do we require the next node in the order to be able to decode.

When both sum-energy and sum-bandwidth constraints are applied, we have the following theorem, proven in Appendix A.

Theorem 1. *Under a sum-bandwidth constraints, if $P_i = P$ for all i then the solution that minimizes delay also minimizes the sum energy.*

In this setting there is no trade off between energy and delay. The minimum-energy route is identical to the minimum-delay route. We give an example in Section VI.

Per-node bandwidth and transmission PSD constraints are useful for modeling ultra-wideband communication systems. In ultra-wideband systems, available bandwidth and transmit power are determined by frequency regulators [22]. Furthermore, constraints on the spreading factor are imposed by limits on hardware complexity as well as requirements of communications standards [23]. Consequently, a large number of orthogonal channels can be available, with each node being able to use exactly one of them.

5) Alternate Objective Functions: The LP framework can accommodate a number of alternate objective functions. For example, one might minimize the sum-energy

$$\sum_{i=0}^{L-1} \sum_{j=i+1}^{L} A_{i,j} P_i$$

subject to end-to-end delay constraints $\sum_{i=1}^{L} \Delta_i \leq \tau_{tot}$.

Alternatively, for example if multiple unicast sessions were active in parallel, one might be interested in minimizing the time-bandwidth footprint. If there were many active sessions such that the interference could be modeled as a constant level of additional background noise ("interference averaging"), the techniques developed in this paper could be applied. To minimize the time-bandwidth footprint of the systems, subject to energy and delay constraints, one would choose the objective function to be

$$\sum_{i=0}^{L-1}\sum_{j=1}^{L} A_{i,j}.$$

But, in general, the addition of interference (even modeled as noise) would add a term to the denominator inside the logarithm in (1), meaning that the resulting resource allocation problem would not be an LP, and therefore out of the scope of techniques considered herein.

Finally, in the place of the unicast setting on which we focus in this paper, multicasting can also be addressed in the current framework by appropriately adjusting the objective function and constraints. We discuss the multicasting scenario further in Section IV-D.

B. Optimizing transmission order

The use of mutual information accumulation makes the optimum transmission order quite different from the non-cooperative multi-hop route. Because the accumulation of mutual information by each node extends across many time slots, the decoding process can have very high complexity. This makes it impossible to solve for the best transmission order efficiently through dynamic programming. At the same time since in a network of $N+1$ nodes there are $\sum_{i=0}^{N}\frac{(N-1)!}{(N-1-i)!}$ distinct orderings ($> 10^{63}$ for $N=50$), exhaustive search of all orderings quickly exceeds computational capabilities.

In this section, we present a theorem that tells us how to improve the transmission order by exploiting the characteristics of the LP solution obtained in Section IV-A. Consider an arbitrary transmission order. Define

$$\mathbf{x}^* = \left[\Delta_1^*, \ldots, \Delta_L^*, A_{0,1}^*, \ldots A_{0,L}^*, A_{1,2}^*, \ldots, A_{L-1,L}^*\right] \quad (9)$$

to be the optimum solution obtained by the linear program for the order. Denote the optimum decoding delay as $T_L^* = \sum_{i=1}^{L}\Delta_i^*$. The following theorem is proved in Appendix B.

Theorem 2. *If $\Delta_i^* = 0$, use T_L^{**} to denote the optimum decoding delay (under the same energy and bandwidth constraints) of the "swapped" transmission order:*

$$\begin{array}{lll}[0,\ldots,i-2,i,i-1,i+1,\ldots,L] & \textit{if} & i \le L-1 \\ [0,\ldots,L-2,L] & \textit{if} & i = L\end{array}. \quad (10)$$

Then $T_L^{**} \le T_L^*$.

The intuition behind Theorem 2 is illustrated in Fig. 2. A solution to the LP with $\Delta_i = 0$ indicates that either node i decodes at *exactly* the same time as node $i-1$ (never the case in reality) or that, although later in the order, node i can actually decode before node $i-1$. Therefore, swapping the ordering of nodes i and $i-1$ will typically decrease the T_L once the LP for the revised order is solved. If $i = L$ the destination is swapped with the node prior to it in the order.

C. Algorithms for route & resource allocation optimization

We can now state the iterative route optimization algorithm.

Algorithm 1:

1) Start with an initial transmission order.
2) Use the linear program of Section IV-A to solve for the parameters of the minimum-delay solution.

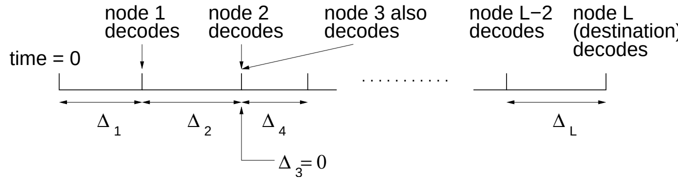

Fig. 2. Intuition behind order-swapping algorithm for $\Delta_3 = 0$.

3) Based on Theorem 2 revise the transmission order:
   a) For any i such that $\Delta_i = 0$ and $\Delta_{i-1} \neq 0$, swap the positions of the two nodes in the order.
   b) If the node $L-1$ is swapped with node L, drop (the former) node $L-1$ from the order entirely. The resulting order contains only $L-1$ nodes.
4) Repeat steps 2)–3) until an ordering is obtained with an associated set of parameters $\mathbf{x}^*$ satisfying $\Delta_i^* > 0$ for all i. At this point terminate the algorithm.

Since the number of constraints in the LP is linear in network size, and the swapping algorithm is very simple, the routing algorithms can usefully be applied to very large networks. While in general we obtain a local minimum, for small networks (of, e.g., 15 nodes, where exhaustive search or orderings is feasible) we found that in our simulations we almost always reach the global optimum. In addition, a number of different initializations can be tried to avoid particularly bad local minima.

In the following sub-sections we discuss various aspects of the algorithm in more depth, such as initialization and characteristics of certain special cases.

1) Initialization: If we initialize Algorithm 1 with an arbitrary transmission order at the target energy constraint(s) we typically find that $\Delta_i^* = 0$ for too many nodes for the search of the order space to get started. To address this issue we introduce the following algorithm that starts from a feasible transmission order and (perhaps) relaxed energy constraint corresponding to that order. Following the presentation of Algorithm 2 we specify the choices we make in various cases.

Algorithm 2:

1) Initialize the algorithm with an initial transmission order and corresponding energy constraints.
2) Tighten the energy constraints slightly.
3) Use Algorithm 1 to re-optimize the route under the new energy constraints.
4) If the energy constraint now equals the target energy, terminate the algorithm. Otherwise, using the newly found route, return to step 2).

As with most non-linear iterative optimization routines, the choice of step size, by which the energy constraints in Algorithm 2 are tightened, is important. Ideally, the energy constraints are tightened only enough that a single Δ_i^* becomes equal to zero. This can typically be accomplished by making the increment small or dynamically choosing the increment (i.e., back-off the increment and resolve if multiple Δ_i equal zero). We now discuss the initial transmission order we use for the per-node and sum-bandwidth scenarios.

When per-node bandwidth constraints (8) are used, then unlike the scenario described by Theorem 1, there is a trade-off between energy and delay. At one extreme, when the energy constraint is fully relaxed, nodes are allowed unlimited energy consumption and the network can thereby achieve the minimum possible transmission delay. The transmission order at this extreme is what we term the *flooding order*, which is easily found as follows. The source node starts transmitting at time 0. Other nodes join in and begin transmitting as soon as they decode. All nodes continue to transmit until the destination decodes. The flooding order and corresponding energy can then be used to initialize Algorithm 2.

In contrast, when a sum-bandwidth constraint is imposed, the flooding order cannot be used to initialize the system. This is because whenever a new node comes on-line in the flooding order the bandwidth usage increases and the sum-bandwidth constraint may be violated. Instead, for these networks we construct our initial transmission order starting from the non-cooperative shortest-path route. If nodes do not perform mutual information accumulation, and if nodes only receive in the time-slot immediately preceding the time at which they decode, then it is easy to solve for the optimum such non-cooperative path using the Dijkstra Algorithm [25]. As our initial transmission order, we add to this shortest-path route the nodes that are able to decode the packet when non-cooperative shortest-path routing is used and all other nodes use mutual information accumulation. We calculate the energy used by this route and initialize the energy constraint accordingly.

2) Characteristics of final route: Since, as noted in the discussion surrounding Thm. 2, nodes will never in reality decode at *exactly* the same time, and since there are only a finite number of orderings, our algorithm will converge. Thus, the mechanism that keeps our algorithm from necessarily reaching the global optimum is the swapping of nodes out of the transmission order. That is, when the $L-1$st node is swapped with node L (the destination), it no longer enters the LP formulation. This makes the decoding constraint (4) easier to meet. Intuitively, it makes sense to drop nodes that are further from the source than is the destination. However, it may turn out that a node that was swapped out of the transmission order could have ultimately prove useful. Our algorithm does not reintroduce nodes and so can converge to a sub-optimal solution.

Because of the exponential number of orderings we expect the problem of finding the optimal transmission order to be NP-hard. Note that for a special case of our problem, namely the low SNR limit where mutual information accumulation and energy accumulation become identical, Maric and Yates [10], [11] already proved that finding the optimal route is NP-hard. Thus, it is not surprising that there must be a caveat to how well our algorithm performs. However, our empirical observation is that, as long as the solution space is "smooth", as one reduces the energy from that used to initialize the search, one almost always reaches the global optimum. On the other hand, we next provide an example of "non-smooth" conditions where at high energy one route is optimal, and at low energy a very different route is optimal, requiring the participation of nodes that do not decode at higher energies and which are therefore dropped from the transmission order by our algorithm.

The example we consider is a four-node network. Node 0 is the source, node 3 is the destination, and $P_i = 1$ and $W_i = 1$ for all i. Consider the situation where $B = 1$, $W_{\text{node}}C_{0,1} = 7\,\text{bits/sec}$, $W_{\text{node}}C_{0,2} = 5\,\text{bits/sec}$, $W_{\text{node}}C_{0,3} = 4\,\text{bits/sec}$, $W_{\text{node}}C_{1,2} = 0\,\text{bits/sec}$, $W_{\text{node}}C_{1,3} = 4\,\text{bits/sec}$, and $W_{\text{node}}C_{2,3} = 17\,\text{bits/sec}$.

When the system has no energy constraint, the flooding order will be $[0, 1, 3]$. Node 1 decodes at $1/7$ second. Then both the source and node 1 transmit for another $3/56$ second, and the destination then decodes. The transmission duration is $\frac{11}{56} \simeq 0.196$ seconds and the energy consumption is $\frac{1}{7} + 2\frac{3}{56} = 0.25$. Node 2 never decodes in this case.

On the other hand, the minimum energy order is $[0, 2, 3]$. Node 2 decodes at $1/5$ second. The source turns off and node 2 starts transmitting. The destination decodes $(1 - 4/5)/17$ seconds later. Node 1 never decodes in this case. The transmission duration is $\frac{18}{85} \simeq 0.21$ seconds and energy consumption is also 0.21 since only one node transmits at a time.

However, if we initialize using the flooding order, we cannot obtain results matching the minimum energy order. If either only the source transmits, or the source transmits until node 1 decodes and then node 1 transmits by itself until the destination decodes, the transmission duration is 0.25 seconds and the energy consumption is 0.25. In both these cases the energy consumption is identical to the flooding route. Thus, without a way to re-introduce node 2 into the transmission order our algorithm would not obtain the optimum minimum energy solution when initialized with the flooding order.

One can consider heuristics for re-introducing nodes into the decoding order. For example, one might query nodes that have been dropped whether they can decode at the current solution, and if they can, reintroduce them into the transmission order. One can see from the four-node example that since node 2 does not decode when the flooding order is used, use of this particular heuristic does not necessarily result in the optimum minimum-energy route being found.

D. Multicasting

The basic multicasting scenario (sending a common message to all nodes) requires all nodes to decode. The only change required in the various versions of the LP stated in (4)–(7) to yield a multicast solution is that L becomes N.

In contrast to the situation in unicasting, in multicasting nodes are never dropped from the transmission order. The main cause for our algorithm only achieving local rather than global optimality discussed in Sec. IV-C2 is thereby obviated. Therefore, we should nearly always achieve the global optimum using our iterative approach. The one remaining caveat is the step-size: it is important to reduce the energy constraint between LPs in small enough increments that only one Δ_i goes to zero per iteration. In a realistic network this will normally be possible, but in an artificial network it is conceivable that node-to-node gains $h_{i,j}$ will coincidentally have values such that multiple Δ_i go to zero at the same time.

There is also a multicasting problem between unicasting and basic multicasting (frequently also called "broadcasting") where we require some subset of the $N+1$ nodes to decode. This scenario is also easy to incorporate into our framework. One simply never drops any of these (now multiple) "destination nodes" from the transmission order. In term of the LP, node L is the index of the last of these destinations to decode.

E. One-dimensional networks

In this section we develop the results on the one-dimensional network presented in Sec. III. Such networks are unrealistically simple, but their simplicity makes it possible to derive analytical results and insight.

The one-dimensional topology and the monotonic path-loss imply that the minimum energy transmission order is $[0, 1, \ldots, L-1, L]$. Furthermore, the sum-bandwidth constraint implies that only one node is active per time-slot—the ith node only transmits in time slot $i+1$.

Since $P_i = P$ for all i we know by Theorem 1 that the minimum delay route is the minimum energy route. This result is especially apparent for this network. The node closest to the destination that has already decoded also has the best channels to all remaining nodes that have not yet decoded. And, when $P_i = P$ for all nodes, it also has the highest $C_{i,k}$ to those remaining nodes. Thus, not only should that node transmit, under a sum-bandwidth constraint it should be allocated all the bandwidth. Energy is therefore not expended anywhere else and the minimum energy and minimum delay routes are the same. If node PSDs are not all the same, the optimum decoding order remains the same and an LP can be solved to find the optimum $\{A_{i,j}\}$. When the P_i are not all the same, there may be an energy-delay trade off.

The transmission delay can be computed by solving $A_{0,1}C_{0,1} = B$, $A_{1,2}C_{1,2} + A_{0,1}C_{0,2} = B$, and in general

$$\sum_{i=0}^{k-1} A_{i,i+1} C_{i,k} = B \tag{11}$$

for each k, $1 \le k \le N-1$, which we can write as as

$$\begin{bmatrix} C_{0,1} & 0 & \ldots & 0 \\ C_{0,2} & C_{1,2} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ C_{0,N} & C_{1,N} & \ldots & C_{N-1,N} \end{bmatrix} \begin{bmatrix} A_{0,1} \\ A_{1,2} \\ \vdots \\ A_{N-1,N} \end{bmatrix} = \begin{bmatrix} B \\ B \\ \vdots \\ B \end{bmatrix}.$$

Note that since the nodes are equally spaced and have identical transmission PSDs, $C_{0,i} = C_{j,i+j}$. We Let $\mathcal{K}$ denote the lower triangular matrix containing the $C_{i,k}$. As the length of the ith time slot is $A_{i-1,i}/W_{\mathrm{T}}$, the transmission delay τ_{c} is given by

$$\tau_{\mathrm{c}} = \frac{\sum_{i=1}^{N} A_{i-1,i}}{W_{\mathrm{T}}} = \frac{B}{W_{\mathrm{T}}} \times [1 \ldots 1]\, \mathcal{K}^{-1} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

The equation above is the general form of the transmission delay for a finite length one-dimensional network, and was used to derive the results plotted in Fig. 1. When N is large and P is small, such that $N^2 P$ is small, the spectral efficiency

$$C_{i,k} = \log_2\left[1 + \frac{h_{i,k}P}{N_0}\right] = \log_2\left[1 + \frac{N^2}{(k-i)^2 D^2}\frac{P}{N_0}\right]$$

is well approximated as $\log_2 e \, \frac{N^2}{(k-i)^2 D^2} \frac{P}{N_0}$. Since P is small the multi-hop route through every node is optimum both for the non-cooperative and cooperative schemes. The incremental decoding delay incurred by each node in the route is $\Delta\tau_{\mathrm{nc}}$ and the overall delay is $\tau_{\mathrm{nc}} = N\Delta\tau_{\mathrm{nc}}$. The incremental delay is calculated as $B = C_{j-1,j} W_{\mathrm{T}} \Delta\tau_{\mathrm{nc}} \simeq \log_2 e \frac{P}{N_0}\frac{N^2}{D^2} W_{\mathrm{T}} \Delta\tau_{\mathrm{nc}}$, and solving for $\Delta\tau_{\mathrm{nc}}$ gives

$$\Delta\tau_{\mathrm{nc}} = \frac{1}{\log_2 e}\frac{BN_0}{PW_{\mathrm{T}}}\frac{D^2}{N^2}.$$

When nodes accumulate mutual information, the incremental delay is reduced. The decoding constraint of the kth node is $B = \sum_{l=1}^{k} C_{k-l,k} A_{k-l,k-l+1}$. In a large network ($N$ large) the $A_{j,j+1}$ will approach a steady state value for $j \gg 0$. The length of each time-slot will also approach a steady state value $\Delta\tau_{\mathrm{c}}$. For such j, since the node is allocated all bandwidth for duration $\Delta\tau_{\mathrm{c}}$, the corresponding allocation $A_{j,j+1} = \Delta\tau_{\mathrm{c}} W_{\mathrm{T}}$. In the asymptotic limit of N large these time-slots dominate the overall delay. In this regime we calculate $\Delta\tau_{\mathrm{c}}$ as $B = \sum_{l=1}^{k} C_{k-l,k} W_{\mathrm{T}} \Delta\tau_{\mathrm{c}} = W_{\mathrm{T}} \Delta\tau_{\mathrm{c}} \log_2 e \, \frac{PN^2}{N_0 D^2} \sum_{l=1}^{k} \frac{1}{l^2}$. In the limit as N (and k) go to infinity, we have $\sum_{l=1}^{\infty} \frac{1}{l^2} = \frac{\pi^2}{6}$ and

$$\Delta\tau_{\mathrm{c}} = \frac{1}{\log_2 e}\frac{BN_0}{PW_{\mathrm{T}}}\frac{D^2}{N^2}\frac{6}{\pi^2}.$$

The cooperative gain is then calculated as

$$\frac{\tau_{\mathrm{nc}}}{\tau_{\mathrm{c}}} = \frac{N\Delta\tau_{\mathrm{nc}}}{N\Delta\tau_{\mathrm{c}}} = \frac{\pi^2}{6}.$$

V. Distributed algorithms

It is often not desirable or even possible to centralize routing and resource allocation as CSI must be aggregated centrally and the resulting decision dispersed globally. Limitations on centralized solutions are particularly constraining for large and temporally varying networks. We have therefore developed two distributed algorithms that are inspired by the characteristics of our centralized solution. These algorithms require far less CSI, perform mutual information accumulation, and yield performance nearly as good as the centralized algorithms.

These algorithms are distributed in the sense that there is no single node that has to have all information about all the channels. On the other hand, as is true for many modern ad-hoc routing algorithms, control messages do need to be propagated through the network.

A. Distributed Algorithm 1

The first distributed algorithm commences with a direct transmission from source to destination. In an iterative fashion intermediate nodes are added to the route.[3] Specifically, the source transmits a sounding signal. All nodes estimate their channel from the source. The destination replies with a second sounding signal. Nodes then estimate their channel to the destination. Given this pair of CSI measurements each node determines the potential energy savings if it were to join the path. Potential energy savings are calculated as

$$\frac{B}{W_{\mathrm{T}}}\frac{(C_{i,L} - C_{0,L})(C_{0,i} - C_{0,L})}{C_{0,i}C_{0,L}C_{i,L}}.$$

Each node broadcasts this information to the rest of the network using any of the many available contention multiple access schemes. The node with the highest energy saving is chosen to participate. In the next step, the CSI from that node to all other nodes in the network is determined. Again, all

[3] The principle of the algorithm is somewhat similar to the PAR algorithm described in [24].

nodes analyze whether they can save energy by joining the route. The process continues until no further (or very limited) energy savings are possible.

The algorithm is simple and, as we see in Sec. VI, very effective. It does have one drawback. The initial setup of a route takes a long time and requires source-to-destination connectivity. If the source-to-destination pathloss is high, a long sounding signal is required (noise averaging over a long time to obtain good CSI estimates). On the other hand, once a route is set up, changes (due to changing channel conditions) can be done rather efficiently, since the route can be modified without tearing down and rebuilding it from scratch. If the source-to-destination pathloss is too high, a hybrid cooperative/multi-hop strategy could be envisioned where nodes close to the destination cooperate to get the message to an intermediate node (the first hop) which then serves as a new source to forward the message on to the destination.

B. Distributed Algorithm 2

A somewhat simpler algorithm can be implemented as follows. The destination broadcasts a sounding signal and all nodes estimate their channels to the destination. The source starts to transmit the information packet. As each node decodes they transmit their node-to-destination CSI to the destination (or to the other nodes) to determine if their CSI is better than the currently transmitting node. If it is, they take over the transmission, and the previously transmitting nodes ceases to transmit. New nodes replace previous nodes in turn until the message reaches the destination.

Because of the lack of network-wide CSI, the algorithms of this section require the use of rateless codes. This is in contrast to the centralized algorithms which could in principle use block codes with appropriately chosen rates once the length of each time slot is solved for by the final LP.

VI. Numerical details of results

In this section we give detailed numerical results for the algorithms developed in this paper for two-dimensional 50-node networks located in the unit square. For all examples the source node 0 is located at $[0.2, 0.2]$ and the destination node 49 is located at $[0.8, 0.8]$. Remaining nodes are placed randomly according to the uniform distribution. A typical network from this ensemble is shown in Fig. 3. In order to give the reader a strong sense of the relationship between geometry and channel strength we study the case where the channel gain $h_{i,j}$ between node i and node j is deterministically related to the Euclidean distance $d_{i,j}$ between them as $h_{i,j} = (d_{i,j})^{-2}$.

To quantify the performance of our algorithm we establish a baseline non-cooperative strategy for comparison. For the baseline we choose multi-hop. Only one node transmits at each time. The route is selected using Dijkstra's shortest path algorithm [25], and each node listens only to the transmission of the node that immediately precedes it in the route. We also consider a hybrid strategy that uses the Dijkstra-based route but where nodes perform mutual-information accumulation (listening to all previous transmissions instead of just the immediately prior transmission). By studying both we get a sense of the fractional performance improvement due to the use of mutual information accumulation, and that due

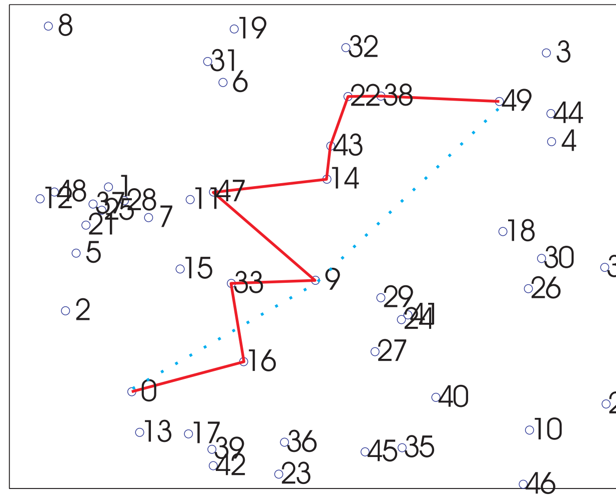

Fig. 3. Location of nodes in fifty node network. The minimum-energy cooperative (solid) and non-cooperative multi-hop (dotted) routes are shown.

to the use of a route designed specifically for cooperative transmission.

Our algorithms were implemented in Python, calling the GLPK [26] LP solver when necessary. We recall from Section IV-C and IV-D that, to aid our search of the order space using Algorithms 1 & 2, we use an adaptive step size when tightening the energy constraint. The ideal tightening is one such that a single $\Delta_i^* = 0$. Say, for instance, that we solve the LP with energy constraint E^+, which is the prior energy constraint reduced by some γ and that we find multiple $\Delta_i^* = 0$. Then, we re-solve the LP with the energy constraint slightly loosened to $E^+ + \gamma/2$. If we still have multiple $\Delta_i^* = 0$ we try again with $E^+ + 3\gamma/4$ and so forth, each time halving the previous increment.

A. System wide bandwidth constraint

We first consider a sum-bandwidth constraint on the specific network shown in Fig. 3 where $B = 28.9$ bits (20 nats), $N_0/2 = 1$, $W_\mathrm{T} = 1$, and $P_i = P = 1$ for all i. By Theorem 1, under a sum-energy constraint the minimum-delay and minimum-energy routes are the same. There is no energy–delay trade off.

After solving for the route using our centralized algorithm, we find that the subset of nodes that actually transmit in the final transmission order is $[0, 16, 33, 9, 47, 14, 43, 22, 38, 49]$, indicated in Fig. 3 by the solid line. As can be seen from inspection of the figure, the nodes that are active in the solution are the nodes that lie closest to the direct path between source and destination. This is due to the fact that channel gain is inversely proportional to distance squared. For this example network the destination decodes after $\tau_\mathrm{c} = 13.09$ seconds.

We now develop results for a non-cooperative multihop routing example. In the non-cooperative case, and as described for linear networks in Section IV-E, the incremental delay accrued by the hop from node i to node j is $B/W_\mathrm{T}\, C_{i,j} = B/W_\mathrm{T} \log_2 \left[1 + \frac{h_{i,j} P}{N_0}\right]$. For the node placements in Fig. 3 the shortest path route is found to be $[0, 9, 49]$, indicated in the figure by the dotted line. The resulting source-to-destination

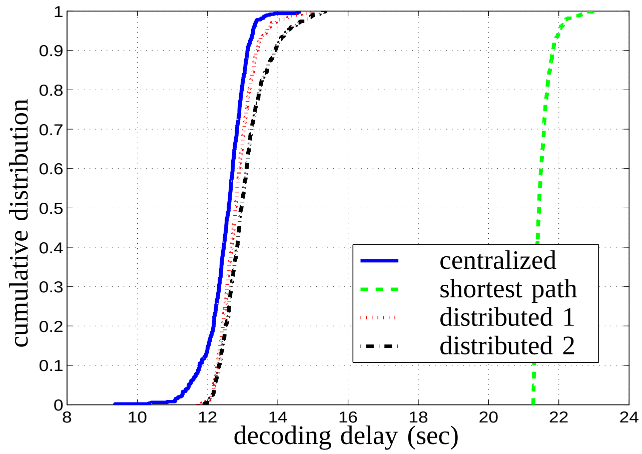

Fig. 4. Delay distribution: centralized, distributed, non-cooperative solutions.

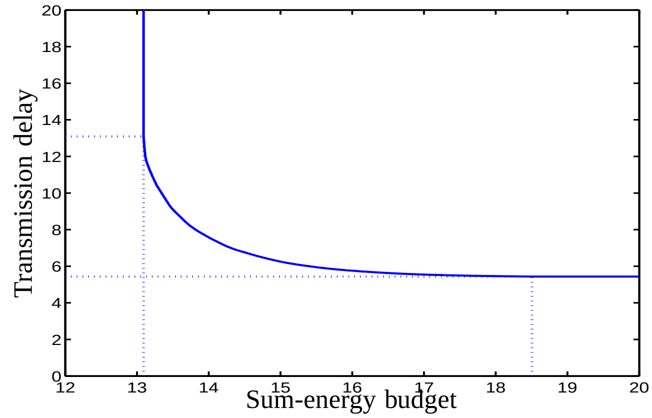

Fig. 5. Delay versus energy trade off in fifty-node network. Nodes are uniformly distributed in the unit square. Channel gains are proportional to d^{-2}, where d is the distance between transmitter and receiver. The sum-energy across all nodes and the per-node bandwidths are limited.

delay $\tau_{\rm nc}$ is 21.47 seconds. There is only one active relay node in the shortest-path route, the one closest to the direct path connecting source to destination. This node (number 9) also participates in the cooperative route.

The decrease in transmission duration obtained by our cooperative route stems both from the use of mutual-information accumulation and the use of a route tuned to cooperation. If the nodes perform mutual information accumulation, but only the nodes in route obtained from Dijkstra's algorithm participate in transmission, the transmission delay is 16.51 seconds. Thus, roughly half the decrease in transmission duration is due to the use of mutual information accumulation, and half due to the use of a route tuned to mutual information accumulation.

To ensure that the improvement is not specific to the sample network of Fig. 3, we calculate the distribution of decoding delays over an ensemble of 500 independently generated realizations of networks of the type depicted in Fig. 3 where the source and destination locations are held constant at $[0.2, 0.2]$ and $[0.8, 0.8]$, respectively, and the rest of the nodes are placed uniformly on the unit square.

The cumulative distribution function (CDF) of decoding delay is plotted in Fig. 4. The average delay of the *centralized* cooperative routing using mutual information accumulation is 12.54 seconds, while the average delay of non-cooperative routing, solved for using Dijkstra's *shortest-path* algorithm, is 21.52 seconds. On average, the conventional non-cooperative multihop transmission incurs additional delay and energy usage on the order of 70%.

In addition, in Fig. 4 we also plot CDF results for the two distributed routing algorithms introduced in Section V. The penalty for using the distributed algorithms in terms of delay (or, equivalently, energy) is small. On average the first distributed algorithm incurs less than 2.5% excess delay as compared to the centralized solution. The excess delay of the second distributed algorithm is less than 4.2%.

B. Per-node bandwidth constraint

We now again consider the network of Fig. 3, but this time under per-node bandwidth constraints. In this setting there is a trade off between system resources (energy and bandwidth) and transmission delay. We keep the same parameters as before, namely $B = 28.9$ bits (20 nats), $N_0/2 = 1$, $P_i = P = 1$, and set the per-node bandwidth constraint $W_i = 1$ for all i. The energy-delay trade off achieved is plotted in Fig. 5.

At one resource extreme we flood the network, fully relaxing the sum-energy constraint and allowing nodes unlimited energy consumption. The network can then achieve the minimum possible transmission delay. In the network depicted in Fig. 3 all nodes except $3, 4,$ and 44 participate in the flooding routing. The order in which nodes come on-line as transmitters is $[0, 13, 17, 39, 42, 16, 2, 36, 23, 15, \ldots, 20, 32, 34, 8, 49]$. The flooding energy is 18.51 and the transmission delay is 5.4.

As the energy budget is decreased, nodes with weaker connectivity to the destination go off-line and only nodes with stronger channels remain active. Finally, at some minimum energy, the network becomes disconnected. The limit point of delay as the energy approaches is defined as the minimum-energy transmission duration. For the network of Fig. 3 the minimum-energy route $[0, 16, 33, 9, 47, 14, 43, 22, 38, 49]$, depicted by the solid line. The minimum energy is 13.09 and the minimum delay is 13.09, the same as in Sec. VI-A. The low-energy route has only a single transmitter transmitting at any given time. This is because if each node waits for all prior transmissions to complete before beginning its own transmission, that node will have accumulated the most mutual information possible. Therefore, the optimum route has only one node at a time transmitting. Since only one node at a time transmits, the system bandwidth is constant. Thus, in the low-energy limit the sum-bandwidth and per-node bandwidth constraints are fully comparable.

When the energy budget is increased, multiple nodes can transmit simultaneously. In contrast, when bandwidth constraints are imposed on a per-node basis, the non-cooperative scheme is limited to the transmission band of a single node. Therefore, the peak bandwidth and energy used by the cooperative strategy when the transmission delay is minimized can exceed that of the non-cooperative strategy, even though the total energy consumption is lower. For instance, for the example discussed in Sec. VI-A, $\tau_{\rm nc} = 21.47$ and since $P_i = 1$ and $W_i = 1$ for all i, the energy consumption of the non-cooperative case is also 21.47, which exceeds the cooperative flooding energy of 18.51 mentioned above. Of course, for this case, the improvement of delay is more impressive: the flooding route has a delay of 5.4 compared to 21.47.

VII. Summary and Conclusions

In this paper we analyze the problem of resource allocation in cooperative relay networks that use mutual-information accumulation. We divided the problem into one of finding the best transmission order and one of finding the best resource allocation given a transmission order. As our solution is based on solving a sequence of LPs it is computationally efficient, even for large networks. We show that under equal per-node PSDs, the minimum-delay solution also minimizes energy consumption. The resulting route is markedly different from the conventional shortest-path route. We develop distributed algorithms that retain most of the performance gains without requiring centralized knowledge of CSI.

The approach of this paper is a step towards the realization of practical cooperative communications in large networks. Future work will focus on optimizing power allocations (the P_i), algorithms that are suitable for imperfect channel state information, and the impact of non-ideal codes and hardware.

Acknowledgements: We thank Dr. Neelesh Mehta for useful discussions, and Dr. Jin Zhang, Dr. Kent Wittenburg, and Dr. Joseph Katz for their support and encouragement.

References

[1] Zigbee Alliance, www.zigbee.org, latest version.

[2] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity-part I: System description," *IEEE Trans. Comm.*, p. 1927–1938, 2003.

[3] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity-part II: Implementation aspects and performance analysis," *IEEE Trans. Comm.*, vol. 51, p. 1939–1948, 2003.

[4] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inform. Theory*, vol. 50, pp. 3062–3080, 2004.

[5] A. Stefanov and E. Erkip, "Cooperative coding for wireless networks," *IEEE Trans. Comm.*, vol. 52, pp. 1470–1476, 2004.

[6] T. E. Hunter, S. Sanayei, and A. Nosratinia, "Outage analysis of coded cooperation,"vol. 52, pp. 375–391, 2006.

[7] R. Madan, N. B. Mehta, A. F. Molisch, and J. Zhang, Energy-Efficient Decentralized Routing with Localized Cooperation Suitable for Fast Fading", *Allerton Conference*, 2007.

[8] G. Barriac, R. Mudumbai, and U. Madhow, "Distributed beamforming for information transfer in sensor networks," in *3rd Int. Symp. Inform. Proc. Sensor Networks IPSN 2004*, pp. 81–88, 2004.

[9] M. Abdallah and H. Papadopoulos, "Beamforming algorithms for decode-and-forward relaying in wireless networks," in *Proc. Conf. Inform. Sciences & Systems (CISS)*, 2005.

[10] I. Maric and R. D. Yates, "Cooperative multihop broadcast for wireless networks," *IEEE J. Selected Areas Comm.*, vol. 22, pp. 1080–1088, 2004.

[11] I. Maric and R. D. Yates, "Cooperative multicast for maximum network lifetime," *IEEE J. Selected Areas Comm.*, vol. 23, pp. 127–135, 2005.

[12] J. Chen, L. Jia, X. Liu, G. Noubir, and R. Sundaram, "Minimum Energy Accumulative Routing in Wireless Networks", *Proc. IEEE Infocom*, pp. 1875–1886, 2005.

[13] J. Castura and Y. Mao, "Rateless coding over fading channels," *IEEE Comm. Lett.*, vol. 10, pp. 46–48, 2006.

[14] A. F. Molisch, N. B. Mehta, J. S. Yedidia, and J. Zhang, "Performance of Fountain Codes in Collaborative Relay Networks," *IEEE Trans. Wireless Comm.*, vol. 6, pp. 4108–4119, 2007.

[15] M.Luby, "LT codes," in *Proc. 43rd Annual IEEE Symp. Foundations of Computer Science(FOCS)*, pp. 271–282, 2002.

[16] J. W. Byers, M. Luby, and W. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," *IEEE J. Selected Areas Comm.*, vol. 20, pp. 1528–1540, 2002.

[17] A. Shokrollahi, "Raptor codes," in *Proc. Int. Symp. Inform. Theory*, p. 36, 2004.

[18] B. Zhao and M. C. Valenti, "Practical Relay Networks: A Generalization of Hybrid-ARQ" *IEEE J. Selected Areas Comm.*, vol. 23, pp.7–18, 2005.

[19] D. S. Lun, N. Ratnakar, M. Medard, R. Koetter, D. R. Karger, T. Hio, E. Ahmed, and F. Zhao, "Minimum-Cost Multicast Over Coded Packet Networks", *IEEE Trans. Inform. Theory,* vol. 20, 2608-2623, 2006.

[20] C. E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*,, vol. 27, pp. 379–423, 623–656, 1948.

[21] M. Mitzenmacher, "Digital fountains: A survey and look forward," in *Proc. IEEE Inform. Theory Workshop 2004*, pp. 271–276, Oct. 2004.

[22] Federal Communications Commission, *First report and order 02-48*, USA (2002).

[23] J. Zhang, P. Orlik, Z. Sahinoglu, A. F. Molisch, and P. Kinney, "Low-Rate and Precision-Ranging UWB Systems", *Proc. IEEE*, vol. 97, pp. 313–331 (2009).

[24] R. Yim, N. B. Mehta, A. F. Molisch, and J. Zhang, "Progressive Accumulative Routing in Wireless Networks", *Proc. IEEE Globecom*, 2006.

[25] T. H. Cormen, C. E Leiserson, and R. L Rivest, *Introduction to Algorithms*, 2nd edition. MIT Press and McGraw-Hill, 2001. Section 24.3, pp. 595-601.

[26] "GNU Linear Programming Kit," http://www.gnu.org/software/glpk.

[27] S. C. Draper, L. Liu, A. F. Molisch, J. S. Yedidia, "Iterative linear-programming-based route optimization for cooperative networks,"in *Proc. Int. Zurich Seminar Commun.*, Zurich, Switzerland, Mar. 2008.

[28] S. C. Draper, L. Liu, A. F. Molisch, J. S. Yedidia, "Routing in cooperative networks with mutual-information accumulation,"in *Proc. IEEE Int. Confi. Commun.*, Beijing, China, May 2008.

Appendix

A. Proof of Theorem 1

Start from the energy used E_{used}

$$E_{\text{used}} = \sum_{i=0}^{L-1}\sum_{j=1}^{L} A_{i,j}P_i = \sum_{i=0}^{L-1}\Delta_i W_{\text{T}} P = T_L W_{\text{T}} P. \quad (12)$$

The second equality must hold else (7) is loose at the optimum. But that would mean that some degrees of freedom A go unallocated in some time slot. If this is the case the decoding time can be strictly decreased by moving up all subsequent decoding times by A/W_{T}.

The third equality holds by definition, $\sum_{i=0}^{L-1}\Delta_i = T_L$. Since the duration of decoding T_L is proportional to the energy used E_{used}, minimizing one minimizes the other.

B. Proof of Theorem 2

Case 1: ($i=1$; the index i refers to the ith non-source node in the transmission order and the ith time slot) Combine node 1's decoding constraint (4) with the total degrees-of-freedom in time slot 1 (7) or (8), for the sum-bandwidth and per-node bandwidth constraints, respectively, to get

$$\frac{B}{C_{0,1}} \leq A_{0,1} \leq \Delta_1^*\, W_{\text{T}} \quad (13)$$

for the sum-bandwidth constraint and

$$\frac{B}{C_{0,1}} \leq A_{0,1} \leq \Delta_1^*\, W_{\text{node}} \quad (14)$$

for the per-node constraint. Eq. (13) and (14) demonstrate for both cases the constraint that no node can decode the message before the source. Therefore, $\Delta_1^* > 0$ is always true (for any ordering) and we need only consider $2 \leq i \leq L$.

Case 2: ($2\leq i\leq L$–1) We show that $\tilde{\mathbf{x}}$ (cf. (9)), a "swapped" version of $\mathbf{x}^*$, is a feasible solution for the swapped ordering that has a decoding delay equal to the optimal decoding delay of the original ordering. Define

$$\tilde{\mathbf{x}} = \left[\tilde{\Delta}_1, \ldots, \tilde{\Delta}_L, \tilde{A}_{0,1}, \tilde{A}_{0,2}, \ldots \tilde{A}_{0,L}, \tilde{A}_{1,2}, \ldots, \tilde{A}_{L-1,L}\right],$$

where

$$\begin{aligned}
\tilde{\Delta}_i &= \Delta_i && \text{for all} \quad i \\
\tilde{A}_{k,l} &= A^*_{k,l} && \text{for all} \quad k, l \text{ s.t. } k \neq i-1, k \neq i \\
\tilde{A}_{i-1,i} &= 0 && \\
\tilde{A}_{i-1,j} &= A^*_{i,j} && \text{for all} \quad j \in \{i+1, \ldots, L\} \\
\tilde{A}_{i,j} &= A^*_{i-1,j} && \text{for all} \quad j \in \{i+1, \ldots, L\}.
\end{aligned}$$

We immediately see $\sum_{i=1}^{L} \tilde{\Delta}_i = \sum_{i=1}^{L} \Delta_i^*$. We now show that $\tilde{\mathbf{x}}$ satisfies all problem constraints.

First note that all but two of the degree-of-freedom allocations $A_{i,j}$ made to each node in each time slot are almost all identical in $\mathbf{x}^*$ and $\tilde{\mathbf{x}}$. There are two exceptions. The first, $A_{i-1,i}$ doesn't appear in $\tilde{\mathbf{x}}$, but $A_{i-1,i} = 0$ since $\Delta_i = 0$. The second, $\tilde{A}_{i-1,i} = 0$.

From this we immediately get that the energy, decoding, and degrees-of-freedom constraints remain satisfied for $\tilde{\mathbf{x}}$. Since the non-zero degree-of-freedom allocations are identical for $\mathbf{x}^*$ and $\tilde{\mathbf{x}}$, the energy usage remains the same under either sum-energy or per-node-energy constraints. For the same reason the decoding ability of nodes $1, \ldots, i-2$, nodes $i+1, \ldots, L$, and the "old" (pre-swapped) node $i-1$ remain unchanged. The old node i doesn't benefit from the old node $i-1$'s transmissions any longer since the order is swapped in $\tilde{\mathbf{x}}$. However, because $\Delta_i = 0$, $A_{i-1,i} = 0$ and it didn't accumulated any mutual information in the old order in any case. Finally, since the positive degree-of-freedom allocations remain the same, and the time-slot durations $\tilde{\Delta}_i$ remain the same, all degree-of-freedom constraints remain satisfied.

Case 3: ($i = L$) For the same reasoning as in case 2, if we define the same vector $\tilde{\mathbf{x}}$, the decoding delay remains the same and all constraints remain satisfied. Now, if we drop the (new) node L from the problem completely (the destination is the new node $L-1$) the reduced solution is still feasible since none of the other nodes relied on the dropped nodes' transmission. (It was the last in the order).